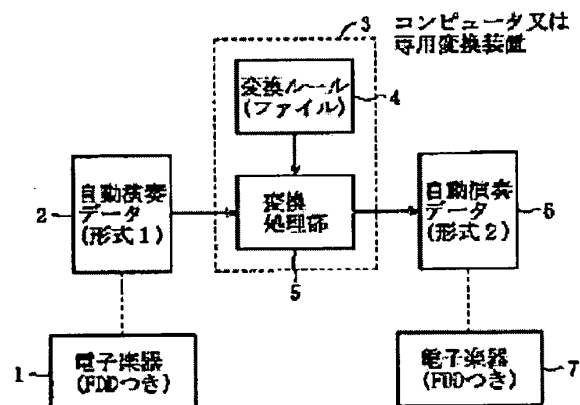


AUTOMATIC PLAYING DATA CONVERTING SYSTEM AND MEDIUM RECORDED WITH PROGRAM

Patent number: JP10124046
Publication date: 1998-05-15
Inventor: KOIKE MASAHIKO
Applicant: YAMAHA CORP
Classification:
 - International: G10H1/00; G10H1/00
 - european:
Application number: JP19970222531 19970819
Priority number(s):

Abstract of JP10124046

PROBLEM TO BE SOLVED: To convert certain format automatic playing data to other format automatic playing data by having the input means which inputs first format automatic playing data and the output means which outputs second format automatic playing data generated by a data converting means.
SOLUTION: First format automatic playing data 2 are converted to second format automatic playing data 6 through a computer 3. The computer 3 has a conversion rule file 4 which stores conversion rules and a conversion processing section 5. The section 5 converts the data 2 into the data 6 in accordance with the conversion rules which are stored in the file 4. Thus, the data 2 are reproduced (played) in an electronic instrument 1, which is for the first format, and the data 6 are reproduced (played) in an electronic musical instrument 7 which is for the second format.



Data supplied from the esp@cenet database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-124046

(43) 公開日 平成10年(1998) 5月15日

(51) Int.Cl.⁶

G 1 0 H 1/00

識別記号

1 0 2

F I

G 1 0 H 1/00

Z

1 0 2 Z

審査請求 未請求 請求項の数 9 O L (全 15 頁)

(21) 出願番号 特願平9-222531

(22) 出願日 平成9年(1997) 8月19日

(31) 優先権主張番号 特願平8-228843

(32) 優先日 平8(1996) 8月29日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000004075

ヤマハ株式会社

静岡県浜松市中沢町10番1号

(72) 発明者 小池 正彦

静岡県浜松市中沢町10番1号 ヤマハ株式会社内

(74) 代理人 弁理士 高橋 敬四郎 (外1名)

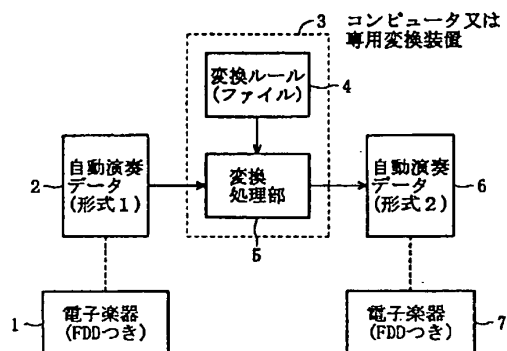
(54) 【発明の名称】 自動演奏データ変換システム及びプログラムを記録した媒体

(57) 【要約】

【課題】 ある形式の自動演奏データから他の形式の自動演奏データに変換することができる自動演奏データ変換システムを提供することを課題とする。

【解決手段】 第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式(2)から第2の形式(6)に自動演奏データを変換するためのルールを記憶する記憶手段(4)と、第1の形式の自動演奏データを入力する入力手段と、入力手段により入力される第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、第1の形式から第2の形式へ自動演奏データを変換するデータ変換手段(5)と、データ変換手段により変換された第2の形式の自動演奏データを出力する出力手段とを有する。

実施例



【特許請求の範囲】

【請求項1】 第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段と、

第1の形式の自動演奏データを入力する入力手段と、
前記入力手段により入力される第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成するデータ変換手段と、

前記データ変換手段により作成された第2の形式の自動演奏データを出力する出力手段とを有する自動演奏データ変換システム。

【請求項2】 第1の形式の自動演奏データを入力する入力手段と、

前記入力手段により入力される第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成するデータ変換手段と、
前記データ変換手段により生成された第2の形式の自動演奏データを出力する出力手段とを有する自動演奏データ変換システム。

【請求項3】 第1の形式の自動演奏データを入力する入力手段と、

前記入力手段により入力される第1の形式の自動演奏データに含まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成するデータ変換手段と、

前記データ変換手段により生成された第2の形式の自動演奏データを出力する出力手段とを有する自動演奏データ変換システム。

【請求項4】 (a) 第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段を準備する工程と、

(b) 第1の形式の自動演奏データを入力する工程と、

(c) 前記入力された第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成する工程と、

(d) 前記作成された第2の形式の自動演奏データを出力する工程とを含む自動演奏データ変換方法。

【請求項5】 (a) 第1の形式の自動演奏データを入力する工程と、

(b) 前記入力された第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成する工程と、

(c) 前記生成された第2の形式の自動演奏データを出力する工程とを含む自動演奏データ変換方法。

【請求項6】 (a) 第1の形式の自動演奏データを入力する工程と、

(b) 前記入力された第1の形式の自動演奏データに含

まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成する工程と、

(c) 前記生成された第2の形式の自動演奏データを出力する工程とを含む自動演奏データ変換方法。

【請求項7】 (a) 第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段を準備する手順と、

10 (b) 第1の形式の自動演奏データを入力する手順と、

(c) 前記入力された第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成する手順と、

(d) 前記作成された第2の形式の自動演奏データを出力する手順とをコンピュータに実行させるためのプログラムを記録した媒体。

【請求項8】 (a) 第1の形式の自動演奏データを入力する手順と、

20 (b) 前記入力された第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成する手順と、

(c) 前記生成された第2の形式の自動演奏データを出力する手順とをコンピュータに実行させるためのプログラムを記録した媒体。

【請求項9】 (a) 第1の形式の自動演奏データを入力する手順と、

(b) 前記入力された第1の形式の自動演奏データに含まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成する手順と、

30 (c) 前記生成された第2の形式の自動演奏データを出力する手順とをコンピュータに実行させるためのプログラムを記録した媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は自動演奏データ変換技術に関し、特にある形式から別の形式の自動演奏データに変換する自動演奏データ変換技術に関する。

【0002】

40 【従来の技術】MIDIデータは、電子楽器共通に使用することができる統一的な演奏情報である。MIDIデータの大まかな規格として、GM規格がある。GM規格は、基本的な演奏情報の規則を決めるものにすぎない。このGM規格の細部をさらに細かく決めたものに、GS規格とXG規格がある。GS規格とXG規格は、共にGM規格に従うものであるが、両者は細部において若干異なる。例えば、両者の音色番号は一部において一致していない。また、効果の種類または効果のパラメータが異なる。

【0003】電子楽器または音源においても、GS規格

用のものとXG規格用のものがある。例えば、XG規格用の電子楽器にGS規格のMIDIデータを供給すると、発音はされるものの、望ましい音色や効果で発音されないことがある。

【0004】一方、近年、通信カラオケが急速に普及している。このカラオケ装置に新しい曲を加えるには、その曲用の自動演奏データを作成し、その自動演奏データを通信によりカラオケ装置に供給する。したがって、曲が増える度に、自動演奏データも増えていく。通信カラオケにおいては、次々に新しい曲用の自動演奏データが作成されているため、既に大量の自動演奏データが世の中に回っている。

【0005】通信カラオケに用いられる自動演奏データは、MIDIデータを基調としたものである。MIDIデータにはGS規格とXG規格の2種類がある。それに対応し、自動演奏データもGS規格のものとXG規格のものの2種類がある。

【0006】例えば、XG規格のカラオケ装置でGS規格の自動演奏データを演奏すると、音色配列の違いなどにより曲の感じが若干異なってしまう、曲本来の演奏を期待することができない。また、ある規格の自動演奏データを大量に保持していても、異なる規格のカラオケ装置では使用できないため、自動演奏データの資源が無駄になってしまう。

【0007】

【発明が解決しようとする課題】MIDIデータまたは自動演奏データの規格が異なれば、電子楽器またはカラオケ装置においてはやや異なる音が発音されてしまい、実質的に使用に耐えられない。また、過去から現在にわたり、大量のMIDIデータまたは自動演奏データを蓄積していたとしても、規格の異なる電子楽器またはカラオケ装置では使用することができず、大量のデータが無駄になってしまう。

【0008】本発明の目的は、ある形式の自動演奏データから他の形式の自動演奏データに変換することができる自動演奏データ変換システム、自動演奏データ変換方法又はプログラムを記録した媒体を提供することである。

【0009】

【課題を解決するための手段】本発明の自動演奏データ変換システムは、第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段と、第1の形式の自動演奏データを入力する入力手段と、前記入力手段により入力される第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成するデータ変換手段と、前記データ変換手段により作成された第2の形式の自動演奏データを出力する出力手段とを有する。

【0010】第1の形式から第2の形式に自動演奏データを変換するためのルールが記憶手段に記憶されている。ユーザまたはメーカは、当該ルールを作成したり変更することができる。ユーザまたはメーカが自由にルールを作成または変更できるので、容易に自動演奏データの変換ルールを定めることができ、自動演奏データの変換の自由度が広がる。また、当該ルールに従い自動演奏データの変換を行えば、数値の変換だけでなく、自動演奏データのデータ列を並び替えるような変換を行うこともできる。ユーザまたはメーカがルールを作成または変更できるので、複雑な変換も可能になる。

【0011】本発明の自動演奏データ変換システムは、第1の形式の自動演奏データを入力する入力手段と、前記入力手段により入力される第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成するデータ変換手段と、前記データ変換手段により生成された第2の形式の自動演奏データを出力する出力手段とを有する。

【0012】自動演奏データ中に含まれる効果付与のデータについても、異なる形式へのデータ変換を行うことができる。第1の形式と第2の形式の自動演奏データの間で、効果付与のデータに互換性がなくても、適正な効果付与のデータに変換することができる。変換された効果付与のデータを含む自動演奏データを、効果付与装置に供給すれば、楽音信号に適正な効果が付与される。

【0013】本発明の自動演奏データ変換システムは、第1の形式の自動演奏データを入力する入力手段と、前記入力手段により入力される第1の形式の自動演奏データに含まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成するデータ変換手段と、前記データ変換手段により生成された第2の形式の自動演奏データを出力する出力手段とを有する。

【0014】第1の形式の自動演奏データにおいては1つのチャンネル情報で足りるが、第2の形式の自動演奏データにおいては複数のチャンネル情報が必要になる場合がある。その場合は、第1の形式の自動演奏データにおける1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成する。複数のチャンネル情報に分割することにより、第2の形式の自動演奏データにおいても適正な自動演奏が可能になる。

【0015】本発明の自動演奏データ変換方法は、

(a) 第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段を準備する工程と、(b) 第1の形式の自動演奏データを入力する工程と、(c) 前記入力された第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成する工

程と、(d)前記作成された第2の形式の自動演奏データを出力する工程とを含む。

【0016】本発明の自動演奏データ変換方法は、

(a)第1の形式の自動演奏データを入力する工程と、
(b)前記入力された第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成する工程と、(c)前記生成された第2の形式の自動演奏データを出力する工程とを含む。

【0017】本発明の自動演奏データ変換方法は、

(a)第1の形式の自動演奏データを入力する工程と、
(b)前記入力された第1の形式の自動演奏データに含まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成する工程と、(c)前記生成された第2の形式の自動演奏データを出力する工程とを含む。

【0018】本発明のコンピュータに実行させるためのプログラムを記録した媒体は、(a)第1および第2の形式の自動演奏データのデータ列を対応付け、第1の形式から第2の形式に自動演奏データを変換するためのルールを記憶する記憶手段を準備する手順と、(b)第1の形式の自動演奏データを入力する手順と、(c)前記入力された第1の形式の自動演奏データに含まれるデータ列を前記ルールに従い並び替え、調整し、第2の形式の自動演奏データを作成する手順と、(d)前記作成された第2の形式の自動演奏データを出力する手順とを含む。

【0019】本発明のコンピュータに実行させるためのプログラムを記録した媒体は、(a)第1の形式の自動演奏データを入力する手順と、(b)前記入力された第1の形式の自動演奏データに含まれる効果付与のデータをデータ変換し、第2の形式の自動演奏データを生成する手順と、(c)前記生成された第2の形式の自動演奏データを出力する手順とを含む。

【0020】本発明のコンピュータに実行させるためのプログラムを記録した媒体は、(a)第1の形式の自動演奏データを入力する手順と、(b)前記入力された第1の形式の自動演奏データに含まれる1つのチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成する手順と、(c)前記生成された第2の形式の自動演奏データを出力する手順とを含む。

【0021】

【発明の実施の形態】図1は、本発明の実施例による自動演奏データ変換システムの全体構成を示す概念図である。

【0022】第1の形式の自動演奏データ2は、コンピュータ3を介して、第2の形式の自動演奏データ6に変換される。コンピュータ3は、変換ルールを記憶する変換ルールファイル4と変換処理部5を有する。変換処理部5は、変換ルールファイル4に記憶されている変換ル

ールに従い、自動演奏データ2を自動演奏データ6に変換する。なお、コンピュータ3は、パーソナルコンピュータ等の汎用コンピュータの他、専用変換装置であってもよい。

【0023】第1の形式の自動演奏データ2は、第1の形式用の電子楽器1において再生(演奏)が可能であり、第2の形式の自動演奏データ6は第2の形式用の電子楽器7において再生(演奏)が可能である。

【0024】第1の形式の自動演奏データ2は、第1の形式用の電子楽器1においては正しく再生(演奏)できるが、第2の形式用の電子楽器7においては正しく再生(演奏)できない。一方、第2の形式の自動演奏データ6は、第2の形式用の電子楽器7においては正しく再生(演奏)できるが、第1の形式用の電子楽器1において正しく再生(演奏)できない。

【0025】電子楽器1と7は、それぞれフロッピディスクドライブ(FDD)を備えており、当該FDDに抜き差し可能なフロッピに自動演奏データ2または6を記憶させることができる。例えば、電子楽器1において使用していた自動演奏データ2を第1のフロッピに記憶させ、当該第1のフロッピをコンピュータ3に挿入する。コンピュータ3は、第1のフロッピに記憶されている自動演奏データ2を読み出し、自動演奏データ6を第2のフロッピに書き込む。自動演奏データ6が書き込まれた第2のフロッピを電子楽器7に挿入すれば、電子楽器7において正しい演奏を行うことができる。

【0026】図2は、図1に示す変換ルールファイル4の内容を示す図である。変換ルールファイル4は、ヘッダ4aと変換ルール4bと変換テーブル4cを有する。ヘッダ4aには、変換元の形式名と変換先の形式名、ファイルの名称、ファイル作成日等の変換に必要なファイルの情報が記載される。

【0027】変換テーブル4cは、メロディ音色(リズム音色を除く)の音色番号を変換するためのメロディ音色変換テーブル4d、リズム音色(メロディ音色を除く)の音色番号を変換するためのリズム音色変換テーブル4e、音色別にベロシティデータ(イニシャルタッチ)を変換するためのベロシティ変換テーブル4f、音色別にボリューム(音量)を変換するためのボリューム変換テーブル4g、およびその他の変換を行うためのその他変換テーブル4hを有する。

【0028】変換テーブル4cは、データ値の変換を行う。データの形式(表記方法)の変換を行うものについては、後に説明する変換ルール4bにより変換を行う。データの値だけを変換するのであれば、変換テーブル4cを用いて容易に変換することができる。例えば、プログラムチェンジ(音色番号)やノートオンイベント中のベロシティ(イニシャルタッチ)は変換テーブル4cで変換を行うことができる。

【0029】変換テーブル4cは、基本的な変換を行う

部分であり、多くの場合に必ず行わなければならない変換である。したがって、メーカーが予め変換テーブル4cを作成することができる。その場合、ユーザは、自己の好みにより、メーカーが作成した変換テーブル4cを編集し変更することもできる。もちろん、ユーザが新たに交換テーブルを作成することもできる。

【0030】変換ルール4bは、テキスト形式の簡易言語で記載することができ、変換テーブル4cでは変換が困難な高度な変換を行うためのものである。例えば、データの形式（記載方法）が異なるものの変換を行う。

【0031】変換ルール4bは、例えば効果付与データの変換を行う。効果には、例えばリバーブ（残響）、フェイザー（位相変調）、ビブラート、ピッチチェンジ等がある。これらの効果は、音源等の機種により、対応可能な効果の数（リバーブやフェイザー等）、または効果*

Data? = length1, data1..., type, length2, data2..., length3, data3...>
... (1)

ここで、? は変換ルールの番号、length1, length2, length3 はそれぞれdata1..., data2..., data3...のデータ列の長さである。typeは、変換の種類を示す。typeの詳細は後に示す。< >内は、省略が可能なフォーマットであることを示す。

【0035】上記の式(1)は、データ列data1...をdata2...に変換するためのルールである。自動演奏データ中にdata1...の列があれば、それをdata2...の列に置き換える。

【0036】typeが0であれば、無条件にdata1...からdata2...への変換を行う。typeが1の場合、基本的にda※

形式1: aa, CH, bb, Eff-type, Param-byte, Param...

形式2: cc, CH, Eff-type(MSB), Eff-type(LSB), Param-byte, Param...

... (2)

ここで、CHは効果を付与するチャンネル番号、Eff-typeは効果のタイプ、Param は効果毎に指定されるパラメータ（可変長）、Param-byteは可変長パラメータのバイト数、aa, bb, ccは定数である。

【0039】形式1の自動演奏データから、形式2の自動演奏データへの変換を行うには、上記の式(1)のフ★

Data1 = 6, aa, REG1, bb, 1, 1, REG2, 0, 6, cc, REG1, 0, 1, 1, REG2:Rev
Tb1

形式1のデータ列data1...は、" aa, REG1(=CH), bb, 1 (=Eff-type), 1(=Param-byte), REG2(=Param) " であり、データ列data1...の長さlength1 は6バイトである。

【0041】形式2のデータ列data2...は、" cc, REG1(=CH), 0(=Eff-type(MSB)), 1(=Eff-type(LSB)), 1(=Param-byte), REG2:RevTb1(=Param) "であり、データ列data2...の長さlength2 は6バイトである。

【0042】式(3)は、効果のタイプとしてリバーブが指定された場合の変換ルールを示す。リバーブは、形

*のパラメータ（リバーブの場合は残響の深さ、ビブラートの場合は振幅と速さ）等が異なる。

【0032】効果付与装置は、規格毎に独自の拡張をした機能を持つことが多い。それに対応し、自動演奏データにおいても、効果に関するデータについては、規格毎にコード（データ）体系が異なっている。

【0033】コード体系が異なる場合には、上記の変換テーブル4cにより変換を行うことが困難であるので、変換ルール4bにより変換を行う。変換ルール4bは、コード体系が異なる場合のように複雑な変換を行うことができる。メーカーまたはユーザは、簡易言語により変換ルール4bを記述することができる。以下に、変換ルール4bのフォーマット例を示す。

【0034】

※ ta1...からdata2...への変換を行う。ただし、data1...からdata2...への変換を正常に行うことができないければ、data1...からdata3...への変換を行う。データ列data3...は、例えばエラーコードである。

【0037】次に、効果データについての変換を行うための変換ルールの具体例を示す。第1の形式（形式1）および第2の形式（形式2）の2種類の自動演奏データがあるとする。各形式のデータ列は、以下の通りである。なお、以下の形式は、現実の電子楽器において用いられているものではなく、仮想のものである。

【0038】

★フォーマットに従い、以下のような変換ルールをテキスト形式で変換ルールファイルに記述すればよい。REG1, REG2はレジスタである。データが変数であるときには、レジスタを用いることができる。同一レジスタは、同じ変数を示す。

【0040】

... (3)

式1ではEff-type=1で表し、形式2ではEff-type(MSB)=0, Eff-type(LSB)=1で表す。

【0043】リバーブのパラメータParam...は、それぞれの値について変換ルールを作成することは効率的ではないので、Param...を変数としてレジスタREG2を用いて変換を行う。パラメータParam...は、ほとんどの場合、数値を変換することで対応できる。ここでは、形式1のパラメータ（例えばリバーブレベル）を、変換テーブルRevTb1によって、形式2のパラメータに変換している。REG2:RevTb1の表記は、REG2の値をインデックスとして

RevTblテーブルを参照することを意味する。

【0044】式(3)では、type=0としている。これは、形式1から形式2へ無条件に変換することを意味する。

【0045】なお、type=1とすれば、形式1のデータ列data1から形式2のデータ列data2への変換が正常に行うことができない場合、形式1のデータ列data1から別のデータ列data3に変換することができる。

【0046】例えば、両者の自動演奏データ間において論理的に扱えるパート数が異なる場合がある。両者のパート数が同じであれば、変換は問題なく行うことができる。しかし、パート数が異なる場合には、一部において割り当てることができないパートが生じる。その場合、割り当てられなかったパートについては、その箇所を特定することができるように、演奏に影響のない特殊データ(エラーコード)を埋め込むように変換することができる。ユーザは、コンピュータまたはシーケンサを用いて、特殊データを検索することにより、その箇所を見つけ出し、編集することができる。

【0047】以上のように、テキスト形式の簡易言語で変換ルール4bを記述することにより、変換テーブル4cだけでは変換することが困難または不可能なデータ列についても、変換することができる。変換テーブル4cおよび変換ルール4bを作成することにより、複雑な変換が可能になる。

【0048】例えば、音色番号の変換を行うには、コード体系を変えずに数値を変換するのみであるので、変換テーブル4cを用いて変換を行うことができる。一方、効果データ等の変換を行うには、コード体系の変換を必要とするため、変換ルール4bを用いて変換を行う。効果データ等は、変換が複雑であり、変換テーブル4cだけでは変換することができない。変換ルール4bを使えば、効果データ等についても変換を行うことができる。

【0049】また、変換ルール4bでは、レジスタREG1、REG2またはテーブルRevTblを用いることにより、効率的に変換ルールを記述することができる。

【0050】図3は、本実施例による自動演奏データ変換システムを実現するためのハードウェアの構成を示す図である。

【0051】バス16には、CPU11、ROM12、RAM13、フロッピディスクドライブ(FDD)14、ハードディスクドライブ(HDD)15、操作子17、および表示器18が接続される。

【0052】フロッピディスクドライブ14またはハードディスクドライブ15は、変換元の自動演奏データと変換後の自動演奏データを記憶する。両者の自動演奏データがフロッピに記憶されている場合は、図1に示す電子楽器1と7において使用することもできる。

【0053】表示器18には、フロッピディスクドライブ14またはハードディスクドライブ15に記憶されて

いる自動演奏データのファイルを一覧表として表示することができる。ユーザは、ファイルの一覧表の中から、操作子17を用いてデータ変換を行いたいファイルを選択することができる。そして、当該ファイルを変換したい変換後の自動演奏データの形式(規格)を操作子17を用いて選択または入力することができる。

【0054】ROM12は、コンピュータプログラムおよび各種パラメータを記憶している。CPU11は、ROM12に記憶されているコンピュータプログラムに従い、フロッピディスクドライブ14等の制御またはデータ変換等の処理を行う。RAM13は、レジスタ、フラグ、バッファを含むCPU11のワーキングエリアを有する。

【0055】変換ルールファイル4は、フロッピディスクドライブ14、ハードディスクドライブ15、ROM12、またはRAM13に記憶される。データ変換処理を高速に行うためには、変換ルールファイル4をRAM13またはROM12に記憶することが好ましい。ただし、変換ルールの書き換えを行う場合は、RAM13に変換ルールを記憶することが好ましい。

【0056】変換ルールを長期保存するには、フロッピディスクドライブ14またはハードディスクドライブ15に変換ルールを記憶し、データ変換時だけRAM13にデータ転送をすればよい。

【0057】HDD15は動作プログラムや自動演奏データ、変換ルールファイル等の各種データを記憶しておく記憶装置である。ROM12に動作プログラムが記憶されていない場合、このHDD15内のハードディスクに動作プログラムを記憶させておき、それをRAM13に読み込むことにより、ROM12に動作プログラムを記憶している場合と同様の動作をCPU11にさせることができる。このようにすると、動作プログラムの追加やバージョンアップ等が容易に行える。CD-ROM(コンパクトディスク・リード・オンリ・メモリ)ドライブ21は、CD-ROM32に記憶されている動作プログラムや各種データを読み出す装置である。読み出した動作プログラムや各種データは、HDD15内のハードディスクにストアされる。動作プログラムの新規インストールやバージョンアップ等が容易に行える。なお、このCD-ROMドライブ21以外にも、外部記憶装置として、光磁気ディスク(MO)装置等、様々な形態のメディアを利用するための装置を設けるようにしてもよい。

【0058】通信インターフェース22はLAN(ローカルエリアネットワーク)やインターネット、電話回路等の通信ネットワーク41に接続されており、該通信ネットワーク41を介して、サーバコンピュータ31と接続される。HDD15内に上記動作プログラムや各種データが記憶されていない場合、サーバコンピュータ31からプログラムやデータをダウンロードするために用い

られる。クライアントとなる本自動演奏データ変換システムは、通信インターフェース22及び通信ネットワーク41を介してサーバコンピュータ31へと動作プログラムやデータのダウンロードを要求するコマンドを通信する。サーバコンピュータ31は、このコマンドを受け、要求された動作プログラムやデータを、通信ネットワーク41を介して本自動演奏データ変換システムへと配信し、自動演奏データ変換システムが通信インターフェース22を介して、これらプログラムやデータを受信してHDD15に蓄積することにより、ダウンロードが完了する。

【0059】なお、本実施例は、本実施例に対応する動作プログラムや各種データをインストールした市販のパーソナルコンピュータ等によって、実施させるようにしてもよい。その場合には、本実施例に対応する動作プログラムや各種データを、CD-ROMやフロッピディスク等の、パーソナルコンピュータが読み込むことができる記憶媒体に記憶させた状態で、ユーザーに提供してもよい。そのパーソナルコンピュータ等が、LAN、インターネット、電話回線等の通信ネットワークに接続されている場合には、通信ネットワークを介して、動作プログラムや各種データ等をパーソナルコンピュータ等に提供してもよい。

【0060】図4は、図3に示したCPU11が行う処理のメインルーチンのフローチャートである。

【0061】ステップS1では、レジスタ、フラグおよびバッファ等の初期化を行う。例えば、音色バンク番号を0に初期設定する。通常、音色指定は、バンク番号（バンクセレクト）とプログラム番号（プログラムチェンジ）との組合せにより指定する。しかし、バンク番号なしでプログラムチェンジだけで音色指定がされた場合は、バンク番号が0であるとみなす。バンク番号0は、スタンダードなメロディ音色を示すバンクである。

【0062】ステップS2は、変換元の自動演奏データファイル（以下、変換元ファイルと呼ぶ）を指定する。表示器には、フロッピディスクドライブまたはハードディスクドライブに記憶されている自動演奏データファイルの全ファイルが表示される。ユーザは、その中から変換を行いたいファイルを選択子により選択する。

【0063】ステップS3では、変換後の自動演奏データの形式を指定する。形式は、例えばGS規格やXG規格である。表示器に全ての形式を表示し、その中からユーザが操作子により選択することができる。また、ユーザが操作子により形式名を入力してもよい。

【0064】ステップS4では、変換元の自動演奏データの形式を検出する。形式は、例えばXG規格またはGS規格である。自動演奏データ中のあるイベントを読み出し、そのイベントの形式を判別することにより自動演奏データの形式を検出することができる。

【0065】例えば、XG規格の自動演奏データの先頭

には、XG規格用のリセット信号が含まれており、GS規格の自動演奏データの先頭には、GS規格用のリセット信号が含まれている。そのリセット信号を判別することにより、自動演奏データの形式がXG規格であるのかGS規格であるのかを識別することができる。

【0066】ステップS5では、前ステップで変換後の形式のイベントが検出されたか否かをチェックする。前ステップでは変換元の形式を検出している。変換元の形式が、ユーザが指定した変換後の形式と同じであれば、変換を行う必要はなく、又、無理に変換を行った場合、異常なデータを生成してしまう可能性がある。

【0067】変換元の形式が変換後の形式と同じであれば、ステップS11へ進み、変換が行われなかった旨を表示器に表示し、ユーザ（利用者）に知らせる。その後、メインルーチンの処理を終了する。

【0068】一方、変換元の形式が変換後の形式と異なれば、データ変換を行うため、ステップS6へ進む。

【0069】ステップS6では、チャンネル情報テーブルを作成する。具体的には、変換元ファイルから自動演奏データを読み出し、使用しているチャンネル番号を調べる。そして、チャンネル情報テーブルに、変換元の自動演奏データにおいて使用しているチャンネル番号を記録する。

【0070】ステップS7では、変換後のデータを記録するファイル（以下、変換後ファイルと呼ぶ）を新規作成する。ここでは、変換後ファイルにヘッダ部分のみが書き込まれる。自動演奏データについては、後の処理により当該ファイルに書き込まれる。

【0071】ステップS8では、変換元ファイルよりイベントを読み出す（取り出す）。ステップS9では、読み出したイベントのデータ変換処理を行う。変換元の形式のイベントは、変換後の形式のイベントに変換され、変換後ファイルに書き込まれる。この処理の詳細は、後に図5および図6のフローチャートを参照しながら説明する。

【0072】ステップS10では、変換元の自動演奏データの全イベントについて変換が終了したか否かをチェックする。全イベントについて終了していないときには、ステップS8へ戻り、次のイベントについての処理を繰り返す。

【0073】全イベントについて終了したときには、メインルーチンの処理を終了する。変換元の自動演奏データは、全て変換後の自動演奏データに変換される。

【0074】図5および図6は、図4に示すステップS9のイベント変換処理の詳細を示すフローチャートである。

【0075】自動演奏データにおいて、音色は、大きく分けて、以下の2種類がある。

- 〔1〕メロディ音色
- 〔2〕ドラム音色

また、上記の音色についてのイベントは、通常、それぞれ以下の順番で並ぶ。

- 【0076】(1) バンクセレクト
- (2) プログラムチェンジ
- (3) ノートイベント

また、それ以外の情報（例えば効果）も必要に応じ、存在する。

【0077】まず、最初にメロディ音色の場合の処理を説明し、その次にドラム音色の処理を説明する。

【0078】〔1〕メロディ音色

(1) バンクセレクト

バンクセレクトのイベントが変換元ファイルから読み出された場合を説明する。読み出されたイベントがバンクセレクトの場合は、ステップS21、S41、S61を介して、ステップS62へ進む。

【0079】ステップS62では、バンクセレクトによって指定されるバンク番号を記憶する。バンク番号は、内部レジスタに記憶されるだけであり、変換後ファイルには記録されない。その後、メインルーチンの処理（図4）へ戻る。

【0080】音色番号は、バンク番号（バンクセレクト）とプログラム番号（プログラムチェンジ）の組み合わせにより指定される。プログラム番号は0から127までであり、128種類の音色に相当する。バンク番号を変えれば、異なるプログラム番号の音色（128音色）が存在する。つまり、バンク数×128の音色が存在する。バンク番号は、MSB（最上位バイト）とLSB（最下位バイト）の2バイトにより指定される。

【0081】(2) プログラムチェンジ

プログラムチェンジが変換元ファイルから読み出された場合を説明する。読み出されたイベントがプログラムチェンジであれば、ステップS21を介して、ステップS22へ進む。

【0082】ステップS22では、バンクセレクトにより指定されたバンク番号がドラムバンクであるかメロディバンクであるかをチェックする。ドラムバンクは、ドラム音色が入っているバンクであり、メロディバンクはメロディ音色が入っているバンクである。例えば、バンク番号MSBが16進数表現で7Fのときドラムバンクである。それ以外は、メロディバンクである。ただし、ある音色が、ある形式の自動演奏データではドラムバンクに属し、他の形式の自動演奏データではメロディバンクに属することもある。

【0083】ここでは、メロディ音色の場合を示す。メロディ音色の場合は、バンクがドラムバンクではないので、ステップS23へ進む。

【0084】ステップS23では、記憶されているバンク番号（S62）と、プログラムチェンジにより指定されたプログラム番号に応じて、変換後のバンク番号とプログラム番号を得る。

【0085】具体的には、メロディ音色変換テーブル4d（図2）を用いて、変換元のバンク番号とプログラム番号を基に、変換後のバンク番号とプログラム番号を得る。メロディ音色変換テーブル4dは、両者の数値（バンク番号とプログラム番号）を対応付けて記憶している。

【0086】ステップS24では、記憶されているバンク番号（S62）と、指定されたプログラム番号に応じて、その音色に応じたベロシティテーブルを得る。ベロシティは、鍵盤の場合、イニシャルタッチ（押鍵速度）、すなわち音量に相当する。ベロシティテーブルは、タッチカーブに相当する。一般的に、押鍵速度が速いほど大きな音が発音される。

【0087】イニシャルタッチは、鍵盤のタッチ感によるものであり、鍵盤（鍵盤を製造したメーカー）が異なればイニシャルタッチ（ベロシティ）も異なる。したがって、自動演奏データの形式が異なれば、ベロシティも異なる。

【0088】そこで、音色別ベロシティ変換テーブル4f（図2）を用いて、変換元のベロシティを基に、変換後のベロシティを得る。ベロシティ変換テーブル4fは、音色別に両者のベロシティを対応付けて記憶している。

【0089】ステップS25では、記憶されているバンク番号（S62）と、指定されたプログラム番号に応じて、その音色に応じたボリューム情報を得て、ボリュームオフセットイベントを生成する。ボリュームは、ベロシティとは異なり、全体的な音量（平均的な音量レベル）を示す。ボリュームオフセットイベントは、ボリュームのオフセット値を設定するためのイベントである。

【0090】ボリュームは、自動演奏データの形式によりそれぞれの主張が出やすい。すなわち、形式により、ボリュームが異なりやすい。例えば、ある会社の音源では他の音色に比べピアノが大きいめのボリュームに設定されている様な場合、自動演奏データもそれを考慮して作成されている。

【0091】そこで、音色別ボリューム変換テーブル4g（図2）を用いて、変換元のボリュームを基に、変換後のボリュームを得る。ボリューム変換テーブル4gは、音色別に両者のボリュームを対応付けて記憶している。

【0092】なお、ベロシティとボリュームは、両方とも音量情報であるので、両者を一体として変換を行ってもよい。ただし、上記のように、ベロシティとボリュームはその性質が異なるので、分けて変換を行った方が、変換テーブルの編集がしやすい。

【0093】ステップS26では、同一チャンネルの情報として、バンク番号（バンクセレクト）、プログラム番号（プログラムチェンジ）、ボリュームオフセット情報を交換後ファイルに記録する。これにより、交換後フ

10

20

30

40

50

ファイルには、変換後の音色情報と音量情報が書き込まれたことになる。その後、メインルーチンの処理(図4)へ戻る。

【0094】なお、音色は、バンク番号とプログラム番号の組み合わせで指定されるが、バンク番号が前のものと変わらない場合は、プログラム番号(プログラムチェンジ)だけで指定することができる。ただし、変換元の自動演奏データにおけるあるバンク番号中の音色は、変換後の自動演奏データにおいては2以上のバンク番号にまたがっている場合がある。その場合に対応するため、このステップではバンク番号とプログラムチェンジの両方を記録する。

【0095】また、ベロシティテーブル(S24)は、後に説明するが、ノートイベントが読み出されたときに、ノートイベント中のベロシティを補正する際に用いられる(S51)。

【0096】(3)ノートイベント

ノートイベントが変換元ファイルから読み出された場合を説明する。読み出されたイベントがノートイベントであれば、ステップS21、S41を介して、ステップS42へ進む。

【0097】ステップS42では、ノートイベントのチャンネルがドラムアサインチャンネルであるか否かをチェックする。つまり、そのノートイベントがアサインされたチャンネルがドラム音色に設定されたものであるか否かをチェックする。具体的には、後に説明するステップS32においてドラムアサインチャンネルのリストがテーブルとして作成され、そのテーブルを参照することによりドラムアサインチャンネルか否かをチェックする。

【0098】ここでは、メロディ音色の場合を示す。メロディ音色の場合は、チャンネルがドラムアサインチャンネルではないので、ステップS51へ進む。

【0099】ステップS51では、ノートイベント中のベロシティを、ステップS24で得たベロシティテーブルによって補正し変更した後、変更したベロシティを含むノートイベントを変換後ファイルに記録する。

【0100】ベロシティテーブルは、ステップS24において音色に応じたテーブルとして得られている。ノートイベントは、ベロシティを含む。ノートイベント中のベロシティを、当該ベロシティテーブルに応じて変換し、変換したベロシティを含むノートイベントを変換後ファイルに記録する。その後、メインルーチンの処理(図4)へ戻る。

【0101】(4)その他(例えば効果付与)

その他のイベントは、例えば効果付与のイベントである。効果付与のイベントは、例えば上記の式(2)に示すデータ列で表される。そして、式(3)に示す変換ルールにより、形式1から形式2への変換を行うことができる。

【0102】その他のイベントが変換元ファイルから読み出された場合を説明する。読み出されたイベントがその他のイベントであれば、ステップS21、S41、S61を介して、ステップS71へ進む。

【0103】ステップS71では、読み出した(受信した)イベントのデータ列を各変換ルールのdata1部と比較する。変換ルール4bは、図2に示す変換ルールファイル4中に記憶されている。また、変換ルールは、上記の式(1)に示すフォーマットにより記述される。式

(1)中のdata1は、変換元のデータ列である。変換ルールファイル4には、1または2以上の変換ルール4bが記憶されているので、各変換ルールのdata1部と比較し、対応する変換ルールを探す。

【0104】ステップS72では、data1部が一致する変換ルールがあるか否かをチェックする。一致する変換ルールがないときには、ステップS81へ進み、読み出したイベントを無視または変換せずに、メインルーチンの処理(図4)へ戻る。なお、一致する変換ルールがないときには、変換を行うまたは行わないを指定できるようにしてもよい。

【0105】一方、一致する変換ルールがあるときには、変換ルールに従い変換を行うため、ステップS73へ進む。

【0106】ステップS73では、データ列data1をtypeに基づいてデータ列data2またはdata3に置き換え、変換後ファイルに記録する。例えば、式(3)に示す変換ルールを基に、式(2)に示す形式1から形式2への変換を行い、変換後ファイルに記録する。その後、メインルーチンの処理(図4)へ戻る。

【0107】なお、変換ルールにおいては、式(3)に示すように、レジスタREG1、REG2またはテーブルRevTblを用いて、変数や配列を参照し、効率的に変換を行うことができる。

【0108】以上は、メロディ音色の処理を説明した。次に、ドラム音色の処理を説明する。

【0109】(2)ドラム音色

(1)バンクセレクト

バンクセレクトは、上記のメロディ音色の処理と同じである。

【0110】ドラム音色は、メロディ音色と異なり音高の指定が必要ない音色である。例えば、バスドラムは単一の音高のみを発音し、音高を指定することはない。そのため、ドラム音色は、メロディ音色とは音色の指定方法が異なる。

【0111】メロディ音色は、バンク番号(バンクセレクト)とプログラム番号(プログラムチェンジ)により音色を指定し、ノートイベント中のノートナンバにより音高が指定される。ドラム音色の場合、バンク番号とプログラム番号によりドラムセットが指定され、ノートイベント中のノートナンバにより当該ドラムセット中のバ

ートが指定される。ドラム音色は、実質的にバンク番号とプログラム番号とノートナンバにより音色が指定される。

【0112】(2) プログラムチェンジ

プログラムチェンジが変換元ファイルから読み出された場合を説明する。読み出されたイベントがプログラムチェンジであれば、ステップS21を介して、ステップS22へ進む。

【0113】ステップS22では、バンクセレクトにより指定されたバンク番号がドラムバンクであるか否かをチェックする。ここでは、ドラム音色の場合を説明する。ドラム音色の場合は、ステップS31へ進む。

【0114】ステップS31では、読み出したプログラムチェンジに対応するプログラム番号を内部レジスタに記憶する。プログラム番号は、内部レジスタに記憶され、変換後ファイルには書き込まれない。プログラム番号は、後にノートイベントが読み出された際に、変換後ファイルに書き込まれる(S45)。

【0115】ステップS32では、図4のステップS6で作成したチャンネル情報テーブルを参照し、ドラム音色のチャンネルを分割し、分割した複数のチャンネルを確保し、プログラム番号とチャンネル番号との対応テーブルを作成する。

【0116】チャンネル情報テーブルは、変換元ファイルで使用しているチャンネル番号を記憶しているテーブルである。プログラム番号とチャンネル番号の対応テーブルは、ステップS42において対象チャンネルがドラム音色のチャンネルかメロディ音色のチャンネルかを判断する際に用いられる。

【0117】変換元ファイルでは、1つのプログラムチェンジ(プログラム番号)に対応し1つのチャンネルが確保されるが、変換後ファイルでは、それに対応し複数のチャンネルを確保する場合がある。その理由を説明する。

【0118】ドラム音色の場合は、バンク番号とプログラム番号によりドラムセットを指定する。一度ドラムセットを指定すれば、その後ドラムセットを変更することは稀である。ドラムセットを指定した後は、ノートナンバを変えることによりドラムのパートを選択する。

【0119】しかし、変換元の形式では、全てのパートが、あるドラムセットの中に含まれているとしても、変換後の形式においてはそれらのパートが複数のドラムセットにまたがって含まれている場合がある。その場合、パートが変わる度に、毎回ドラムセットを変更する方法も考えられるが、ドラムセットが変更されると現在発音中のドラムセットの音が途中で変更されてしまうこともありえる。望ましくは、ドラムセットを変更しても、発音中の音は前のドラムセットの音を発音し続けることである。

【0120】上記の問題点を解決するため、変換元の形

式で1つのドラムセットが指定された場合、変換後の形式ではそれに対応し、必要に応じて複数のドラムセットを発音するための複数のチャンネルを確保する。ただし、変換元の形式のドラムセットが、変換後の形式ではドラム音色とメロディ音色の両方を含む場合は、ドラムセットとメロディ音色を発音するための複数のチャンネルを確保する必要がある。

【0121】変換元の形式と変換後の形式が分かっているならば、あるドラムセットが指定されれば、最低確保すべきチャンネル数を予め知ることができる。例えば、変換元の形式においてあるドラムセットが指定されれば、4つのドラムセットと2つのメロディ音色を発音するため、6つのチャンネルを確保すればよい。これは、変換元のあるドラムセットに含まれるパートは、変換後においては6つの音色セットにまたがっていることを意味する。言い換えれば、変換元のチャンネル情報を複数のチャンネル情報に分割し、変換後のチャンネル情報を作成する。

【0122】ステップS33では、前ステップでチャンネル数が足りなくて必要なチャンネル数を確保できなかった場合、確保できなかったチャンネルの情報を対応テーブルに記録する。後に、当該チャンネルを用いたノートイベントが読み出されたときは、発音することができない旨を示すコードを含めて変換後ファイルに書き込むことができる。当該チャンネルのノートイベントは、発音することができないが、シーケンサまたはコンピュータで編集することにより、所望の対応をすることができる。上記の対応テーブルを記録した後、メインルーチンの処理(図4)へ戻る。

【0123】(3) ノートイベント

ノートイベントが変換元ファイルから読み出された場合を説明する。読み出されたイベントがノートイベントであれば、ステップS21、S41を介して、ステップS42へ進む。

【0124】ステップS42では、ノートイベントのチャンネルがドラムアサインチャンネルであるか否かをチェックする。ドラムアサインチャンネルか否かは、ステップS32で作成したテーブルを基に判断する。ここでは、ドラム音色の場合を説明する。ドラム音色の場合は、チャンネルがドラムアサインチャンネルであるので、ステップS43へ進む。

【0125】ステップS43では、記憶されているバンク番号(S62)と、プログラム番号(S31)と、指定されたノートイベント中のノートナンバ(ノート番号)に応じて、変換後のバンク番号、プログラム番号、ノートナンバを得る。

【0126】具体的には、リズム音色変換テーブル4e(図2)を用いて、変換後のバンク番号、プログラム番号、ノートナンバを得る。

【0127】ステップS44では、記憶されているバン

ク番号(S62)と、プログラム番号(S31)と、指定されたノートイベント中のノートナンバ(ノート番号)に応じて、その音色に応じたベロシティテーブルを得る。具体的には、音色別ベロシティ変換テーブル4f(図2)を用いて、変換後のベロシティテーブルを得る。

【0128】ステップS45では、変換後のプログラムチェンジ(プログラム番号)に基づいて対応テーブル(S33で作成したテーブル)を参照し、チャンネルのアサインが可能であるか否かをチェックする。アサインが可能であれば、そのチャンネルの情報として変換したバンク番号、プログラム番号、ノートナンバと、変換したベロシティを変換後ファイルに記録する。

【0129】対応テーブルには、ドラム音色のチャンネルを分割する際に確保することができなかったチャンネルの情報が記録されている。当該対応テーブルを参照することにより、チャンネルのアサインが可能か否かを判断することができる。チャンネルのアサインが可能であれば、変換したバンク番号、プログラム番号等を記録し、その後、メインルーチンの処理(図4)へ戻る。

【0130】ドラム音色の場合は、バンク番号とプログラム番号によりドラムセットが指定され、ノートナンバにより当該ドラムセット中のパートが指定される。

【0131】(4)その他(例えば効果付与)
その他のイベントは、上記のメロディ音色の処理と同じである。なお、ステップS32で作成した対応テーブルを参照し、ドラム音色がアサインされているチャンネルにピッチベンド等により効果が付与される場合は、ドラム音色用に分割されたすべてのチャンネルについて効果を付与する。

【0132】本実施例によれば、メロディ音色だけでなく、ドラム音色についても自動演奏データの変換を行うことができる。変換テーブルを用いれば、数値変換のようなデータ変換を行うことができる。例えば、音色やベロシティである。さらに、変換ルールを用いれば、変換テーブルでは変換が不可能または困難なものであっても、所望の変換を行うことができる。ユーザは、テキスト形式の簡易言語で変換ルールを記述することができるので、ルールの作成が容易である。

【0133】ユーザまたはメーカーは、変換ルールまたは変換テーブルを作成したり変更することができる。ユーザまたはメーカーが自由にルールを作成または変更できるので、容易に自動演奏データの変換のためのルールを定めることができ、自動演奏データの変換の自由度が広がる。

【0134】また、自動演奏データ中に含まれる効果付与のデータについても、異なる形式へのデータ変換を行うことができる。変換元と変換後の自動演奏データとの間で、効果付与のデータに互換性がなくても、適正な効果付与のデータに変換することができる。変換された効

果付与のデータを含む自動演奏データを、効果付与装置に供給すれば、楽音信号に適正な効果が付与される。

【0135】さらに、ドラム音色のように、変換元の自動演奏データにおいては1つのチャンネル情報で足りるが、変換後の自動演奏データにおいては複数のチャンネル情報が必要になる場合がある。その場合は、変換元の自動演奏データにおける1のチャンネル情報を複数のチャンネル情報に分割し、変換後の自動演奏データを生成する。複数のチャンネル情報に分割することにより、変換後の自動演奏データにおいても適正な自動演奏が可能になる。メロディ音色であってもドラム音色であっても、適正な発音が可能である。

【0136】本実施例による自動演奏データ変換システムは、カラオケ装置で用いられる自動演奏データ、電子楽器で用いられる自動演奏データ、MIDIデータのみからなる自動演奏データ等を変換することができる。

【0137】なお、変換を行う際には、全ての交換ルールまたはテーブルを記述する必要はなく、あるカテゴリーについては特別な指定がない限りデフォルトの変換を行い、指定があるときに限り変換ルールまたはテーブルにより変換を行うようにしてもよい。

【0138】例えば、各プログラムチェンジについて変換ルールまたはテーブルが存在しないときは、代表の変換ルールまたはテーブルでベロシティテーブルを変換するようにしてもよい。

【0139】また、自動演奏データの変換は、コンピュータで行ってもよいし、従来の電子楽器の内部で行ってもよい。

【0140】さらに、変換を行う前後の自動演奏データは、フロッピー等に記録せずに、ネットワークを介した電子データとして、自動演奏データ変換システムに対し入力または出力してもよい。

【0141】以上実施例に沿って本発明を説明したが、本発明はこれらに制限されるものではない。例えば、種々の変更、改良、組み合わせ等が可能なことは当業者に自明であろう。

【0142】

【発明の効果】以上説明したように、本発明によれば、ユーザまたはメーカーが自由に変換のためのルールを作成または変更できるので、自動演奏データの変換の自由度が広がる。また、当該ルールに従い自動演奏データの変換を行えば、数値の変換だけでなく、自動演奏データのデータ列を並び変えるような複雑な変換を行うこともできる。

【0143】また、第1の形式と第2の形式の自動演奏データとの間で、効果付与のためのデータの値が異なっているとしても、適正な効果付与のデータに変換することができる。効果付与装置においては、変換後の効果付与のデータを基に適正な効果を楽音信号に付与することができる。

【0144】さらに、第1の形式の自動演奏データにおける1のチャンネル情報を複数のチャンネル情報に分割し、第2の形式の自動演奏データを生成することにより、第2の形式の自動演奏データにおいても適正な自動演奏が可能になる。

【図面の簡単な説明】

【図1】 本発明の実施例による自動演奏データ変換システムの全体構成を示す概念図である。

【図2】 図1に示す変換ルールファイルの内容を示す図である。

【図3】 本実施例による自動演奏データ変換システムを実現するためのハードウェアの構成を示す図である。

【図4】 図3に示したCPUが行う処理のメインループのフローチャートである。

*

*【図5】 図4に示すステップS9のイベント変換処理の詳細を示すフローチャートである。

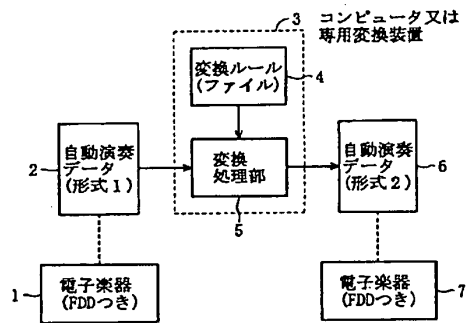
【図6】 図5の処理に続くフローチャートである。

【符号の説明】

1, 7 電子楽器、 2, 6 自動演奏データ、
3 コンピュータ、4 変換ルールファイル、 4 a
ヘッダ、 4 b 変換ルール、 4 c 変換テー
ブル、 5 変換処理部、 11 CPU、 1
2 ROM、 13 RAM、 14 FDD、
15 HDD、 16 バス、 17 操作子、
18 表示器、 21 CD-ROMドライブ、 22
通信インターフェース、 31 サーバコンピュ
ータ、 32 CD-ROM、 41 通信ネットワ
ーク

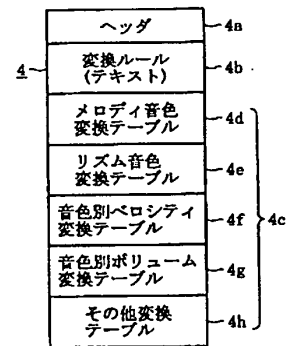
【図1】

実施例

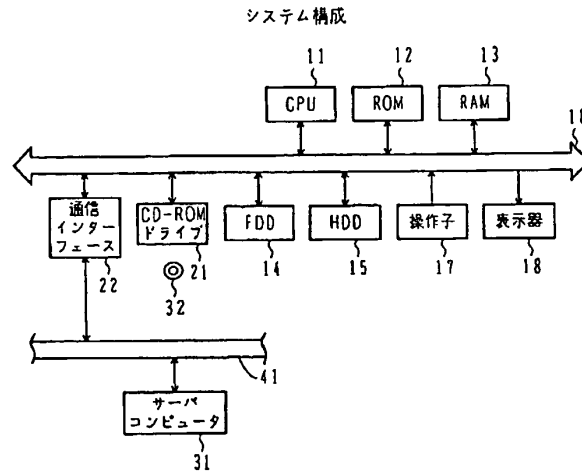


【図2】

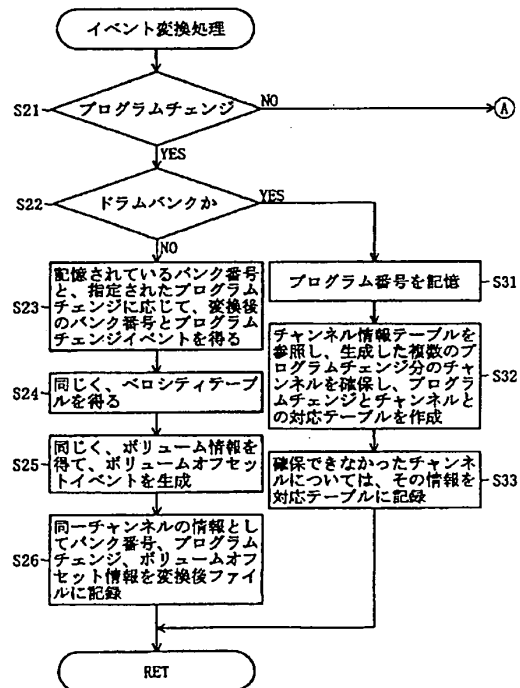
変換ルールファイルの例



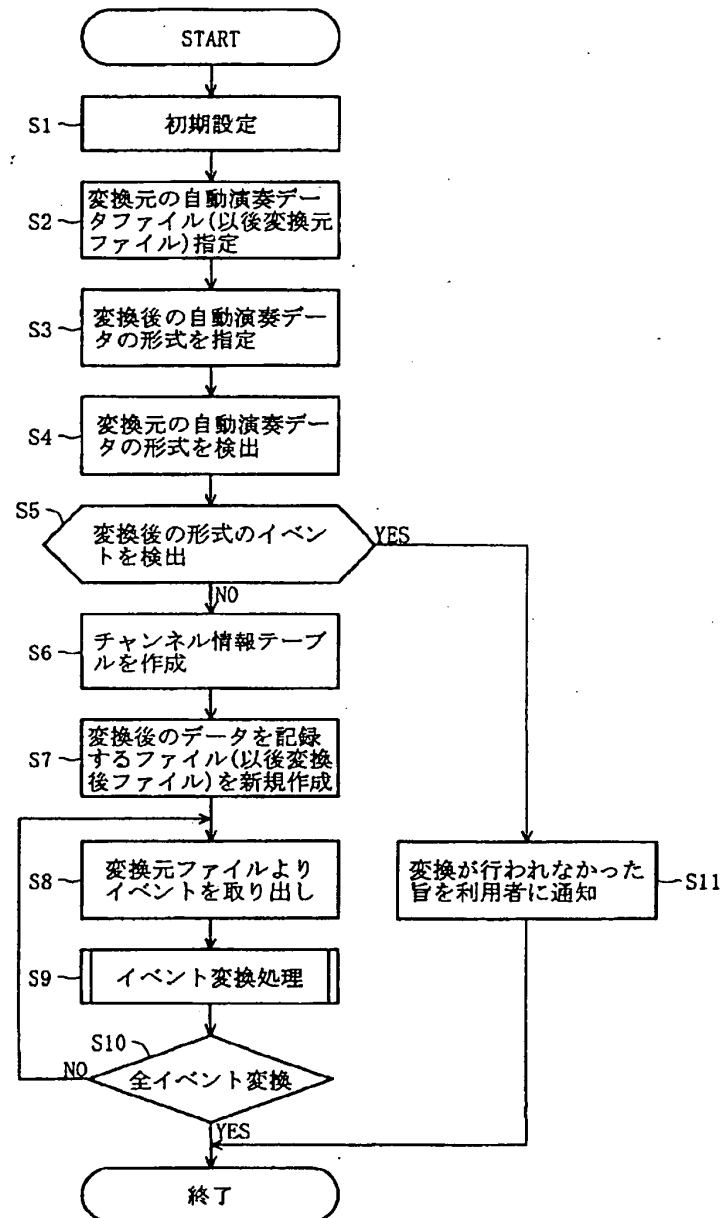
【図3】



【図5】



【図4】



【図6】

